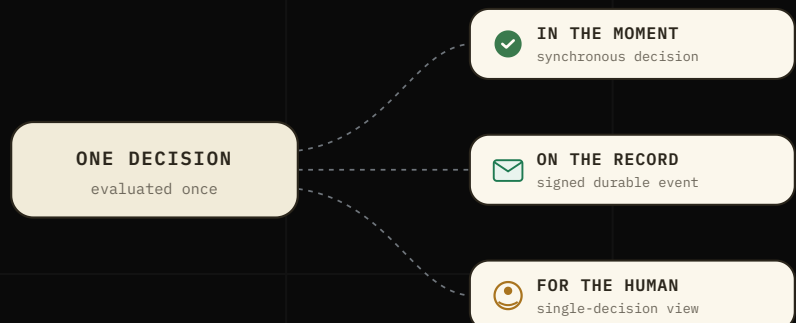




TECHNICAL BRIEF • DECISIONING

Acting on risk decisions: realtime, recorded, reviewable.

A risk decision has three audiences: the transaction flow that must act now, the systems that must record it, and the human who must review it later. Loci exposes a purpose-built surface for each, and all three show the same decision.



one decision, described the same way
act, then record, then review

Acting on risk decisions: realtime, recorded, and reviewable

OVERVIEW

THREE SURFACES

A risk decision has three audiences, and they are not the same. The system carrying the transaction needs an answer in the moment: proceed, hold, or stop.

The systems around it need a durable record of what was decided, so they can route, notify, reconcile, and learn. And a human needs to open any single decision later and understand exactly why it went the way it did.

Most platforms serve one of these well and improvise the other two. Loci exposes a purpose-built surface for each, and all three show the same decision, described the same way.

THE THREAD THAT RUNS THROUGH THEM

Every decision a machine makes should be readable, defensible, and re-runnable by the human who asked for it. Below is how that decision reaches you three times, and what to do with it each time.

You send a transaction; Loci returns a decision before it proceeds

SURFACE

01

IN THE MOMENT

This is the surface that gates the flow. You post the transaction and the context that gives it meaning: the entity, the amount and direction, the timing, and any fields your controls reference.

Loci evaluates the active controls against the entity's baseline behavior and returns a decision fast enough to sit inside the transaction flow rather than beside it. In benchmarking the end-to-end path measured 21 to 49 ms at p95.

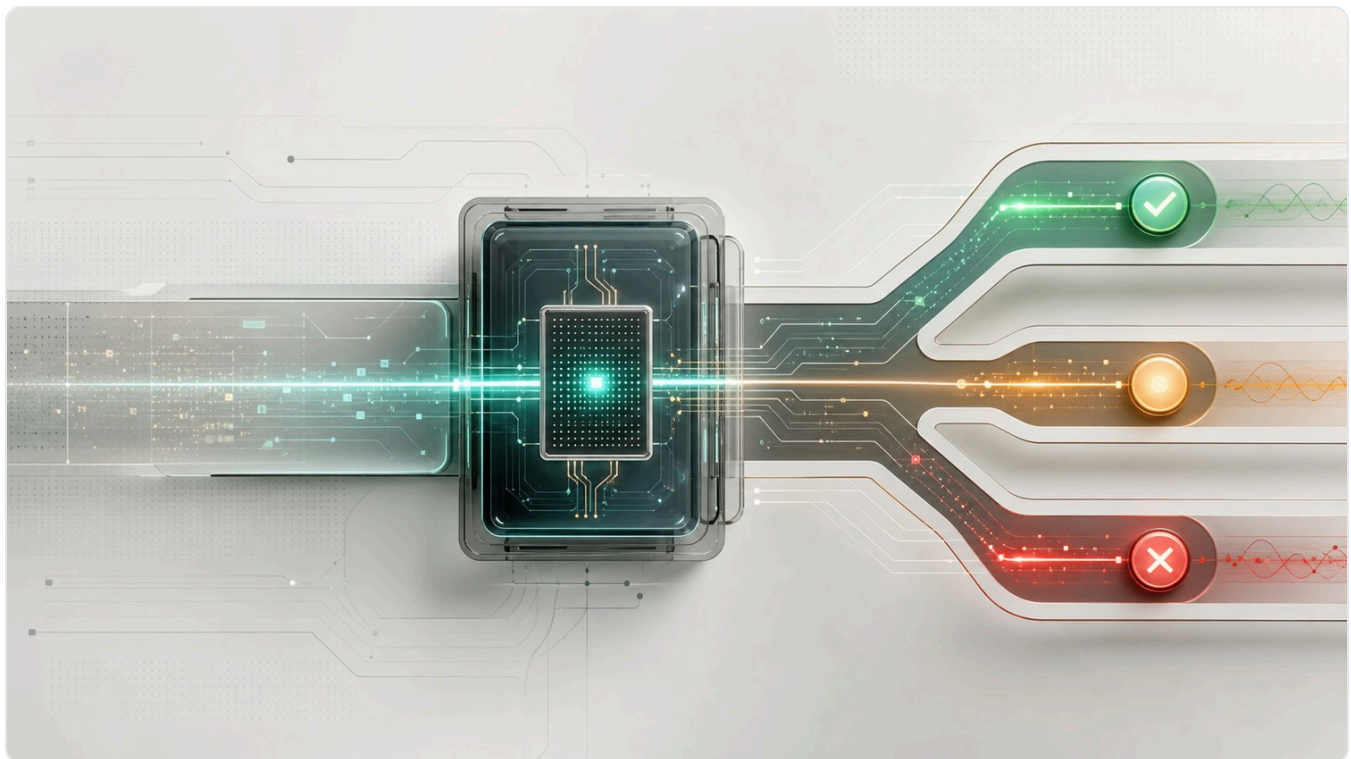


EXHIBIT 1 One evaluation, three gated outcomes. The synchronous response returns approve, review, or decline in time to sit inside the transaction flow.

The fields you act on

FIELD	MEANING	HOW DOWNSTREAM SYSTEMS USE IT
decision	approve, review, or decline	The primary control signal. Let approvals through, route reviews to step-up or a queue, stop declines before settlement.
fraud_score	Composite risk on a normalized 0 to 100 scale	Feed your own tiered handling. A review at 32 is not a review at 71.
decision_reason	Plain-language explanation of the outcome	Log it with the transaction; surface it to whoever or whatever handles a held transaction.
decision_source	Which layer drove it: a blocking action, a critical control, or the score band	Tells you whether a decline came from a hard control or a score threshold, which changes how you treat borderline cases.
data.triggeredRules	The controls that fired, with weights and scores	The evidence behind the number. Attach to the case; step up only when a specific control fired.
policy.thresholds	The approve, review, and decline bands in force	Lets a client render or reconcile the decision against the org's current policy.

The design point: the decision and its reasons arrive together, in one response, fast enough to sit inside the transaction flow. You are not given a score and left to guess. You are given the outcome, the band it fell in, and the named controls that produced it.

BENCHMARK NOTE

The 21 to 49 ms p95 figures are benchmark results, not a contractual service level. They were measured end to end through the gateway on a 6 vCPU configuration at roughly 500 transactions per second, over 60 days of retained history with up to 500,000 transactions per entity, where the control evaluation itself contributed about 4 to 6 ms.

Your own latency depends on control complexity, history depth, deployment topology, and network placement. Treat these as a reference point and validate against your workload.

The synchronous response acts; the event records

SURFACE

02

ON THE RECORD

The synchronous response tells the transaction flow what to do. It does not, on its own, feed everything else that needs to know. That is the webhook's job.

Loci emits a `decision.created` event for every successful evaluation that produces a decision, approvals included, delivered durably with retries and signed so you can trust its origin. A request that fails validation or cannot be evaluated returns an error and produces no decision event.

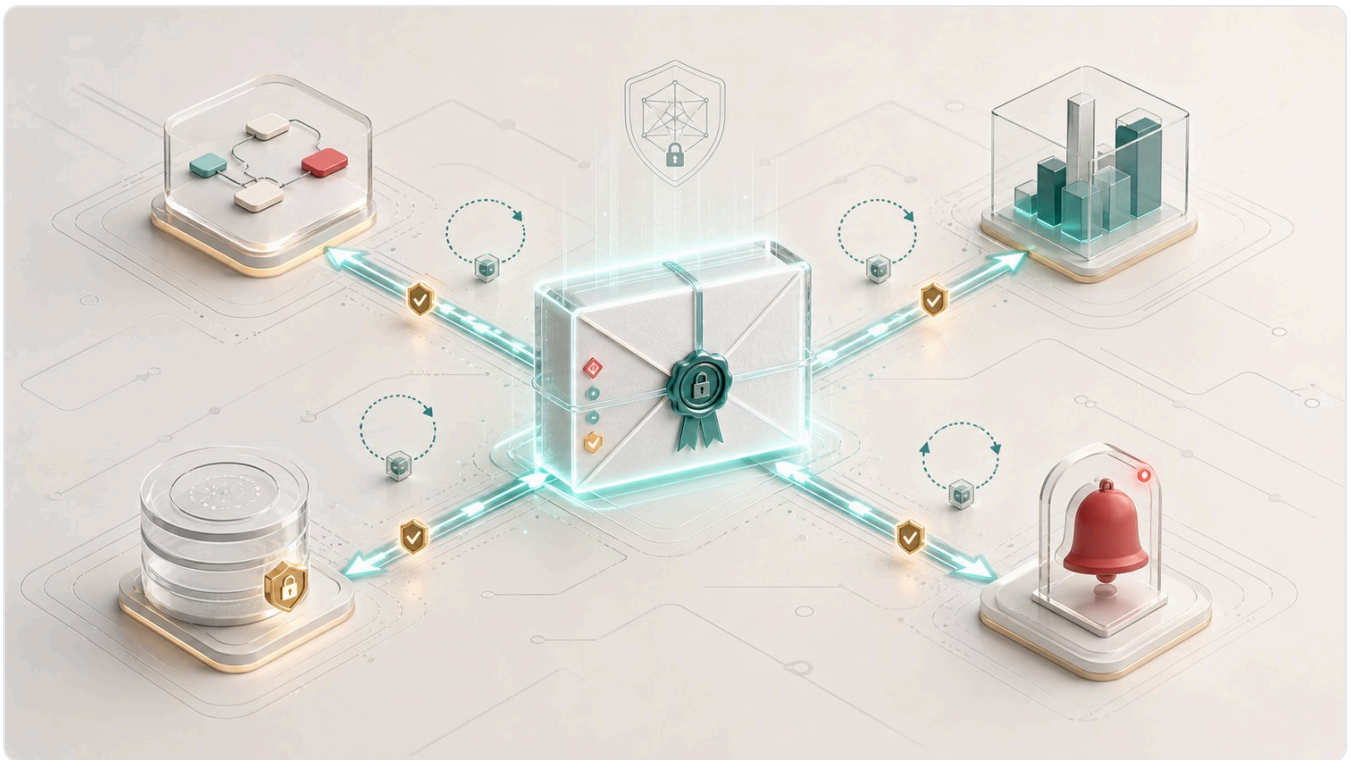


EXHIBIT 2 One signed event, delivered durably to every system that needs the decision on the record: cases, the warehouse, downstream ledgers, and alerting.

Every event shares one envelope

FIELD	MEANING
<code>event_id</code>	Stable id for this event. Use it as your idempotency key so a redelivery is a no-op.
<code>event_type</code>	<code>decision.created</code> . The same channel carries control and screening events under their own types.
<code>event_version</code>	The envelope contract version, so your parser can evolve safely.
<code>occurred_at</code>	When the decision was made.
<code>org_id</code>	The organization the decision belongs to.
<code>data</code>	The decision itself.

Inside `data` you get the same outcome the synchronous response carried (the decision, the score, the reason, whether it was flagged or approved), the controls that triggered, and the transaction as evaluated. Every delivery carries a signature header derived from a secret only you and Loci hold, plus a timestamp, so your endpoint can verify the event is genuine before acting on it.

The distinction from surface one is timing and durability. The synchronous response is for the decision you must act on now. The event is for every system that needs the decision recorded, delivered reliably, and consumed on its own schedule.

WHAT THIS SURFACE IS FOR

Case and workflow systems. Turn a review or decline into a case; the event carries the evidence, so the case opens with the reasons already attached.

Analytics and the warehouse. The event fires on every decision, not only the flagged ones, so your base rates are complete. Approvals are signal too.

Ledgers and downstream records. Annotate the transaction record with the decision and its reason, so your system of record and Loci agree.

Notifications and alerting. Fire on the outcomes that matter to you, filtered to the decision bands you care about.

Not a score, but the case for the score

SURFACE

03

FOR THE HUMAN

Numbers and events serve machines. A person reviewing an alert needs the decision opened up. The control plane's single-transaction view is that surface: it takes one decision and lays out everything behind it on one screen.

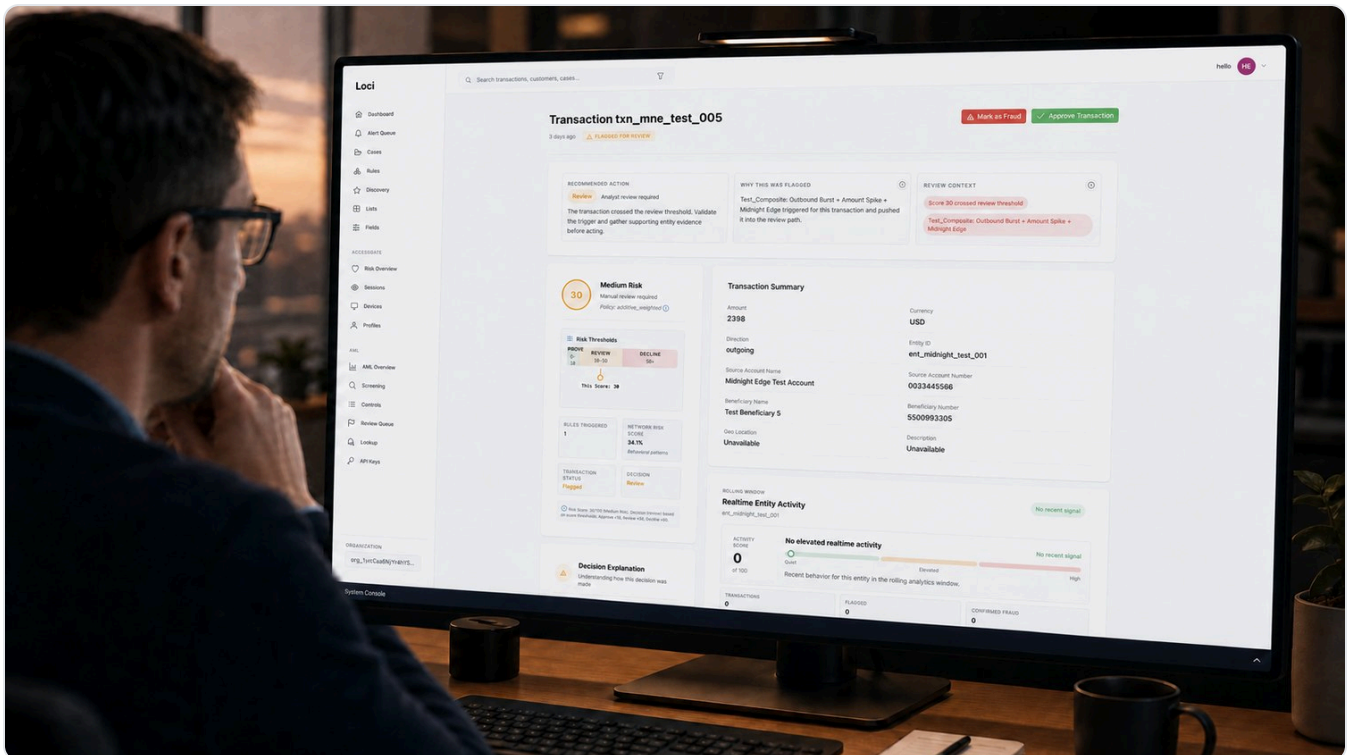


EXHIBIT 3 The single-decision view. The recommended action and the reason it was flagged sit at the top, the score reads against its bands, and the disposition is one click, all on one screen.

What the analyst sees, and works with

Recommended action and why

The action and the reason it was flagged, in plain language, at the top. No hunting.

The controls that fired

Named, so the reviewer knows which behavior triggered the alert rather than inferring it from a total.

The score in its bands

Shown against the org's approve, review, and decline thresholds, so a borderline case reads as borderline.

Recent activity and network risk

The transaction's context. A single transfer means one thing from a dormant account and another from a busy one.

A one-click disposition

Approve or mark as fraud in place, which closes the loop and feeds the record.

This surface is where the platform's promise gets tested, because it is where a human either can or cannot defend the decision.

Every element on the screen traces to a control someone authored and a value the engine computed. So the answer to "why did this happen?" is always available, and the answer to "was this right?" becomes a decision the reviewer records rather than a mystery they inherit.

Three windows onto the same evaluated decision

TOGETHER

ONE TRUTH,
THREE VIEWS

The reason these three surfaces agree is that they are not three systems producing three views. They are three windows onto the same evaluated decision.

The outcome and its reason in your synchronous response, the data in your webhook event, and the summary on the analyst's screen are the same facts, formatted for the moment each one serves: the transaction flow that must act, the systems that must record, and the person who must review.

Field names differ by surface.

The facts underneath do not.

Build against them accordingly

Gate the flow on the synchronous decision. Feed your records, cases, and analytics from the event. Send your analysts to the single-decision view when a human needs to look.

Each surface gives you the outcome, the reasons, and the evidence, because in a system built so that every decision can be read and defended, there is no other kind of output to give.

Loci exists so that every risk decision a machine makes can be read, defended, and re-run by the human who asked for it. Three surfaces onto one decision are how that holds from the transaction flow to the analyst's screen.

SEE THE THREE SURFACES ON YOUR FLOW

A short technical walkthrough: the synchronous decision, the signed event, and the single-decision view a human can defend. Read this brief online at runloci.com/brief/acting-on-risk-decisions.

LOCI FRAUD AI · TECHNICAL BRIEF · JULY 2026

runloci.com · sales@runloci.com